

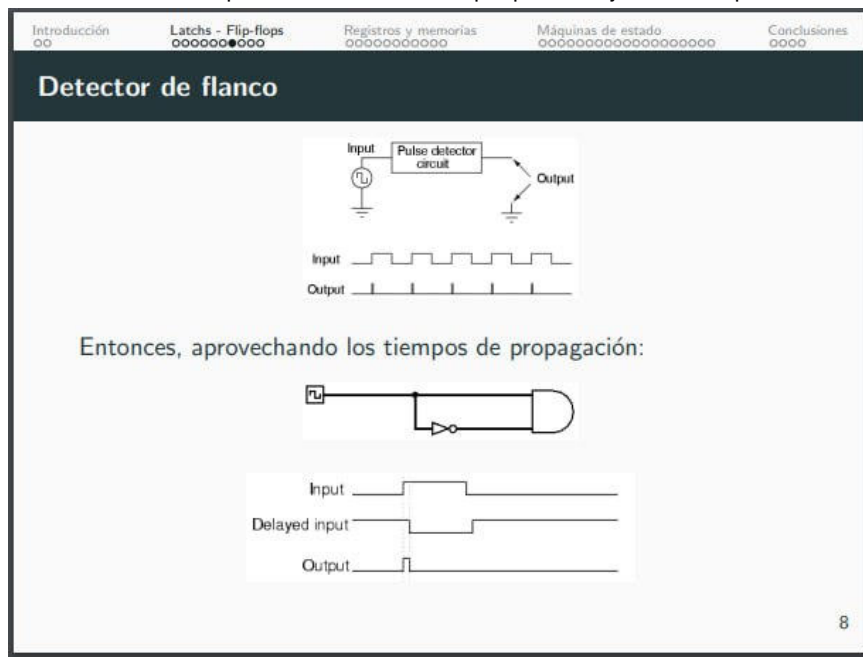
Preguntas del parcial

1. Cómo implementaría un circuito que se comporta de igual manera que una fórmula de lógica proposicional?

Lo implementaría usando compuertas logicas

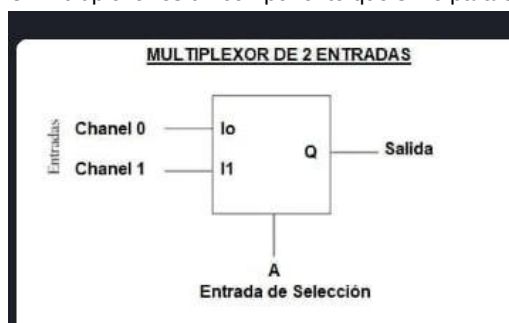
2. ¿Qué función cumple la señal de reloj? ¿Qué circuito permite detectar el flanco de subida de la señal de clock?

La señal del reloj sirve para ordenar cuando debe pasar cada parte del circuito, el flanco ascendente es detectado por el detector de flanco que es un and que tiene de entrada el reloj y el reloj negado, cuando el flanco esta en ascendente sucede que el negador de la señal se encuentra encendido por atraso al mismo tiempo que el reloj causando que el and se active

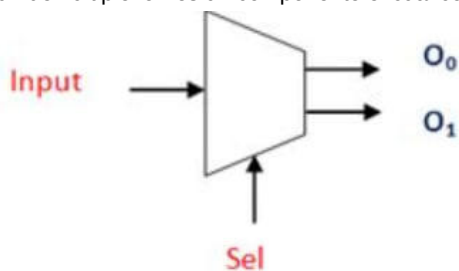


3. ¿Qué es un multiplexor y un demultiplexor?

Un multiplexor es un componente que sirve para elegir que señal dejar salir



un demultiplexor es un componente el cual con una señal elige donde enviarla



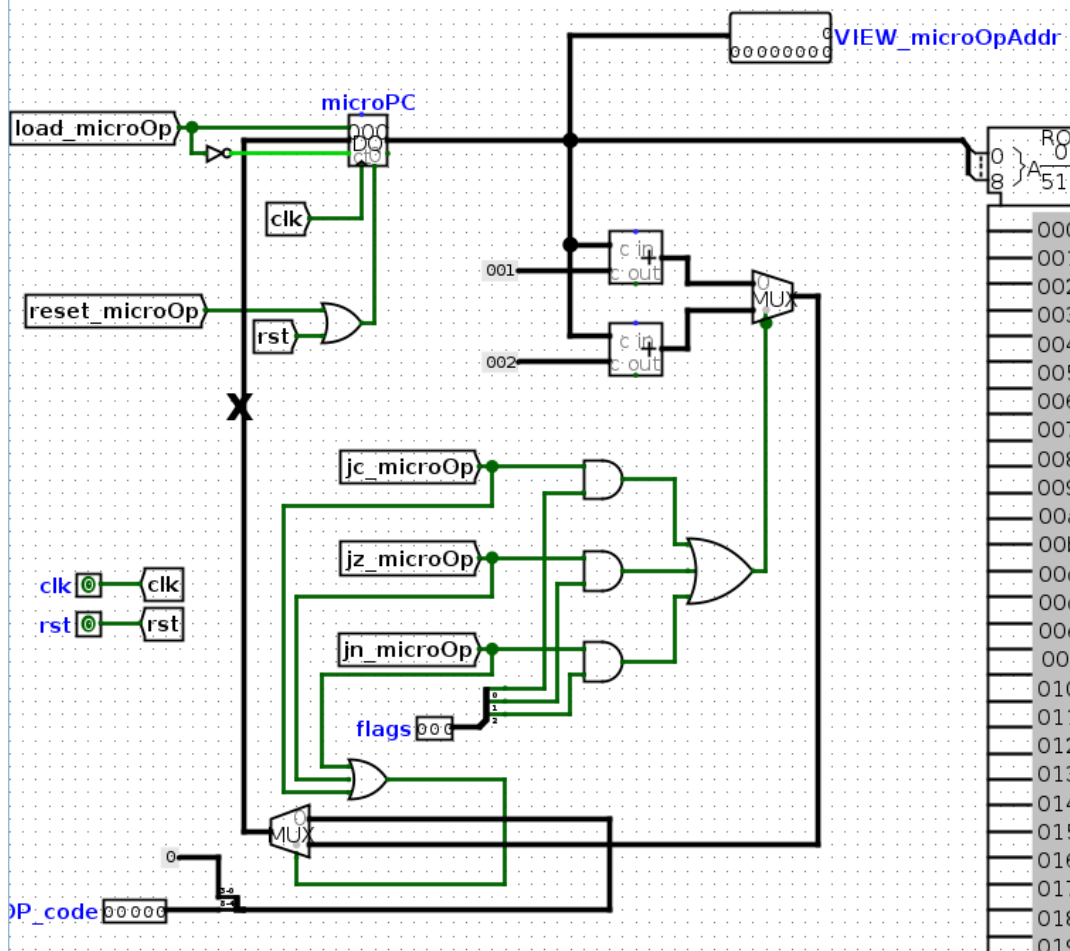
4. ¿A qué le decimos estado de un procesador?

Llamamos estado del procesador a como se encuentran todos los componentes en cada parte del proceso de un programa

5. ¿Qué función cumple la señal load_microop en orga1small? ¿Qué uso le damos al micropc?

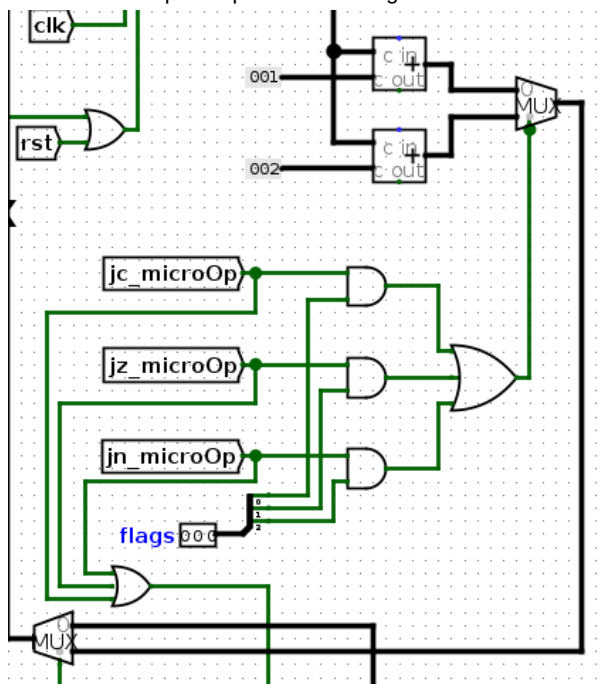
el load_microOp es una señal que solo se utiliza cuando hay saltos ya que sirve para decidir en cuanto incrementar el microPC o cargar el

opcode dependiendo de si se usa en un salto condicional



6. ¿Cuál es el mecanismo que utilizamos en orga1small para implementar los saltos condicionales?

Se usa un multiplexor para con las flags decidir en cuanto aumentar el microPC



7. ¿Qué efecto tiene activar la señal de reset_microop en la unidad de control de orga1small?

La Señal sirve para resetear el microPC

8. ¿Cuál es el mecanismo que utilizamos en orga1small para comenzar a ejecutar una rutina de atención de interrupciones?

Tenemos un controlador de interrupciones que se encarga de verificar si el procesador esta recibiendo interrupciones y si hay interrupciones, cuando se activa el flag i se hace el fetch a la posicion 0x0000 que es donde se guarda la rutina de interrupciones (rai)

9. ¿Cuál es la ventaja de tener operaciones de tres registros en RISC-V en lugar de dos como suele ser el caso en otras arquitecturas?

La ventaja es que no tienes que pisar un registro que estas usando para hacer la operacion para guardar el resultado y que en otra arquitectura si necesitas pisar.

10. En RISC-V, ¿qué significa que un registro sea temporal?

Un registro temporal es aquel que se puede modificar libremente entre llamadas, nada nos asegura que preserven su valor al llamar a una función y retornar.

Machete: convención de llamadas

En general, al llamar a una función hay seis pasos a seguir:

1. Poner los argumentos de la función en los registros para argumentos: a0-a7.
2. Saltar a la función (en RISC-V, esto se hace con jal o jalr, o call).
3. Preservar los registros que haga falta: s0-s11, sp, ra.
4. Ejecutar la función.
5. Poner el resultado en los registros destinados a los valores de retorno: a0-a1.
6. Restaurar los registros que se preservaron en el paso 3.
7. Retornar al programa que invocó la función (jalr, o ret).